一种面向多类不平衡协议流量的改进 AdaBoost.M2 算法

张仁斌，张 杰†，吴 佩

(合肥工业大学 计算机与信息学院, 合肥 230009)

摘 要：针对 AdaBoost。M2 算法在解决多类不平衡协议流量的分类问题时存在不足，提出一种适用于因特网协议流量多类不平衡分类的集成学习算法 RBWS-ADAM2，本算法在 AdaBoost。M2 每次迭代过程中，设计了基于权重的随机平衡重采样策略对训练数据进行预处理，该策略利用随机设置采样平衡点的重采样方式来更改多数类和少数类的样本数目占比，以构建多个具有差异性的训练集，并将样本权重作为样本筛选的依据，尽可能保留高权重样本，以加强对此类样本的学习。在国际公开的协议流量数据集上将 RBWS-ADAM2 算法与其他类似算法进行实验比较表明，相比于其他算法，该算法不仅对部分少数类的 F-measure 有较大提升，更有效提高了集成分类器的总体 G-mean 和总体平均 F-measure，明显增强了集成分类器的整体性能。

关键词：流量分类；集成学习算法；多类不平衡；泛化性能

中图分类号：TP393.04 doi: 10.3969/j.issn.1001-3695.2018.01.0010

Improved AdaBoost.M2 algorithm for multiclass imbalanced protocol traffic

Zhang Renbin, Zhang Jie†, Wu Pei

(*School of Computer & Information*, *Hefei University of Technology*, *Hefei* 230009, *China*)

Abstract: The existing AdaBoost. M2 algorithm are insufficient in protocol traffic multiclass imbalance to solve the problem. So, this thesis proposes an ensemble algorithom called RBWS-ADAM2 for the classification of multiclass internet traffic. During each iteration of AdaBoost. M2, this algorithm preprocessed the training dataset by randomly balanced resampling, this strategy changed the number of majorities and minorities by randomly setting the sampling balance point to build multiple different training datasets. Moreover, this strategy toke sample weight as the basis for sample screening to strengthen the learning of this kind of sample. The experimental comparison of RBWS-ADAM2 algorithm and other similar algorithms on the internationally published protocol traffic datasets shows that, compared to other algorithms, the proposed RBWS-ADAM2 algorithm not only improves the F-Measure of most minorities, but increases the overall G-mean and the overall average F-measure effectively, and obviously enhances the overall performance of the ensemble classifier.

Key words: traffic classification; ensemble algorithm; multiclass imbalance; generalization performance

0 引 言

当前因特网协议流量中存在严重的多类不平衡问题[1]，某些类别的流样本数远多于其他类别。这种情况下，分类器虽然可以取得较高的总体分类精度，但往往偏向于对总体分类精度贡献较大的多数类样本，对少数类样本的分类精度则较低。目前处理数据类别不平衡的方法主要分为两类。一类是在分类器训练之前对训练集进行重采样的方法，其基本思想是通过改变训练集的类别分布来消除或降低数据集不平衡程度，典型的有 SMOTE[2]算法/SBC 算法[3]，SCUT 算法[4]等。以上的重采样算法大都是针对两类不平衡问题进行研究，而文献[5,6]表明针对两类不平衡问题的重采样算法无法有效解决多类不平衡问题甚至可能会带来负面影响，并且部分针对两类不平衡问题的重采样算法无法直接用于解决多类不平衡问题，如前述 SBC 算法，此外对于类似 SMOTE 算法的过采样或欠采样算法，采样比例是影响分类器最终性能的关键因素，但是对于不同数据集，最合适的采样比例可能不同也不容易确定。

另一类是对学习算法改进使之适用于不平衡数据分类，其中最常见的方法是基于 Bagging 的方法和基于 Boosting 的方法，例如将 SMOTE 重采样与 Bagging 集成学习相结合的 SMOTEBagging 算法[7]。还有将重采样方法与 AdaBoost.M2 相结合进行改进的集成学习算法也被相继提出，AdaBoost.M2[8]作为多类集成学习算法，虽然具有相对较优的整体性能，但是其毕竟没有考虑到样本多类不平衡的影响，因此文献[9]提出了

收稿日期：2018-01-16；**修回日期：**2018-03-15

作者简介：张仁斌（1969-），男，湖北武人，副教授，博士，主要研究方向为计算机网络与工业安全；张杰（1993-），男（通信作者），硕士研究生，主要研究方向为计算机网络与信息安全（2250899314@qq.com）；吴佩（1993-），女，硕士研究生，主要研究方向为计算机网络与信息安全.

SMOTEBoost 算法，该算法在基分类器学习之前运用 SMOTE 算法为少数类构建新样本，以提高训练集中少数类的比重，但是该算法的最优过采样比例很难确定，若设置过小则无法达到预期效果，若设置过大则可能导致分类器过拟合，并且过度利用 SMOTE 过采样进行样本扩充，可能带来严重的样本重叠和噪声问题[10]。文献[11]提出了 RUSBoost 算法，该算法在基分类器学习之前运用随机欠采样删除部分多数类样本，该策略可以降低数据的不平衡程度，却可能由于其随机性而误删有用的多数类样本，影响到分类器对多数类的分类效果。还有部分算法不依赖任何重采样算法来平衡数据集，例如文献[12]提出了基于数据分区的集成学习算法，该算法将多数类样本分为若干份，每次训练基分类器时取其中一份与少数类合并作为新训练集，虽然其避免了使用重采样算法，但是每次基分类器只能学习到多数类中的部分样本，可能会降低集成分类器对多数类的分类精度，并且其分区策略只适用于二类不平衡的情况，因此无法很好解决多类不平衡问题。

综上所述，为增强 AdaBoost.M2 算法在面对多类不平衡协议流量时的整体分类性能，缓解多类不平衡对少数类分类的影响，本文提出一种基于随机平衡重采样的改进 AdaBoost.M2 集成学习算法-RBWS（random balance sampling based on weighting）-ADAM2（AdaBoost.M2），本算法通过在 AdaBoost.M2 每次迭代前对训练集进行基于样本权重的随机平衡重采样来提升分类器对于少数类的分类能力，以缓解多类不平衡问题对分类器的影响。在数据预处理过程中，本算法首先随机确定各多数类和各少数类的样本占比，再根据样本占比对训练集进行重采样形成新训练集以增大训练集之间的差异性，有利于提升分类器的泛化性能。此外，在对训练集重采样时，本算法将过采样与欠采样策略相结合，避免了单一使用过采样或欠采样可能导致的不足，同时本算法将样本权重作为唯一的样本筛选标准，优先扩充或保留各类别中高权重样本，进而保证分类器对此类样本的充分学习。

1　RBWS-ADAM2 算法

1.1　基于样本权重的随机平衡重采样策略

为提升 AdaBoost.M2 算法在多类不平衡协议流量数据集上的整体表现，本节设计了针对 AdaBoost.M2 改进的重采样策略 RBWS，该策略包括对所有少数类执行过采样和对所有多数类执行欠采样两个过程，首先通过计算所有类别样本数目的平均值来确定随机区间，并在随机区间内随机确定采样平衡点，再根据采样平衡点来确定多数类和少数类。在对少数类的过采样过程中，为加大对少数类中分类错误样本的关注，对于每个少数类，按照其所有样本的权重大小排序，取前一半样本进行 SMOTE 过采样以生成新样本。在对多数类的欠采样过程中，考虑到网络协议流量存在严重的类内子概念[13]问题，部分样本占比小的子概念可能会被过度采样，所以为保证欠采样后多数类样本的总体质量，本策略采取对各多数类进行基于权重的聚簇式欠采样方法，首先对单个多数类所有样本进行聚簇以得到所有子簇，并按照子簇样本数目占比分配该子簇的欠采样数目，在各子簇中按照样本权重排序进行欠采样，优先删除权重小的样本，以保证分类器对高权重样本的学习。RBWS 重采样策略的伪代码描述如下。

Input:dataset $S=\{(x_1,y_1),?\ ,(x_m,y_m)\}$，$y_i$?$\{1,?\ ,k\}$，$i\{1,?\ ,m\}$．　//k 为类别数，m 为样本总数

Output: new training dataset S'.

M=m/k　//计算所有类别样本数目的平均值 M

d=Random(n_1M, n_2M)　//从随机区间为[n_1M, n_2M]选取随机值 d 作为采样平衡点，其中 0<n_1≤1，1<n_2<[Max/M]，Max 为所有类别的最大样本数目

for each *i* $\{1,?\ ,k\}$ **do:**

　if n[*i*] > d:　//类别 *i* 样本数目 n[*i*]大于 d

　　add *i* to C　//将类别 *i* 加入多数类集合 C

　else: add *i* to U　//将类别 *i* 加入少数类集合 U

end for

for each C_iC **do:**

　num[C_i]=number[C_i]-d　//获取多数类 C_i 的欠采样数目

　SUM_C = SUM_C+num[C_i]　//SUM_C 为所有多数类欠采样的总数目

　clusters=Cluster(C_i)　//对类别 C_i 的所有样本聚簇，得到簇集 clusters

　for each $cluster_j$ clusters **do:**

　　num[$cluster_j$]=number[$cluster_j$]/number[C_i]×num[C_i]

//按样本数目占比确定各子簇的欠采样数目，样本数目越多的子簇需要欠采样的数目越多

　　undersampling $cluster_j$ by weight　//在每个子簇中优先欠采样样本权重小的样本

　end for

end for

get C'?after undersampling　//C' 为多数类欠采样后的样本集

for each U_iU **do:**

　num[U_i]=SUM_C/q　//对所有少数类，过采样数目为 SUM_C 除以所有少数类个数取整，q 为少数类类别数

　get U_i' by weight　//对单个少数类所有样本按照权重大小进行降序排序，取排序后前一半样本 U_i'

　SMOTE(U_i')　//按照 num[U_i]对 U_i' 执行 SMOTE 过采样

end for

get U'?after oversampling　//U' 为少数类过采样后的样本集

return $S'=C'+U'$　//将 C' 与 U' 合并得到新训练集 S'

1.2　RBWS-ADAM2 集成学习算法

研究[14]表明增加训练数据的差异性对于提升集成学习算法的整体性能和解决集成学习算法面临的数据集不平衡问题起十分关键的作用。本节 RBWS-ADAM2 算法基本思想就是在

AdaBoost.M2 每次迭代过程中运用前述 RBWS 重采样策略进行数据预处理，从而在缓解数据不平衡的同时增大训练集的差异性。AdaBoost.M2 算法有两种权重处理方式，分别是 Boosting by reweighting [15]和 Boosting by resampling[15]。Boosting by reweighting 方式要求基分类器可以直接训练带权重的训练集，其优点是所有样本都会参与分类器训练，但是该方式并未考虑多类不平衡的影响，并且由于训练集所有样本都参与训练，每次迭代之间的训练集缺乏差异性，无法有效提升集成分类器的泛化性能。Boosting by resampling 方式的基本思想是每次迭代通过对训练集进行基于样本权重的有放回随机抽样来构造新训练集，该方式虽然对基分类器没有特殊要求，但是可能存在同一样本被重复选入的情况，若某类别中含有过多的困难样本，则采样后的新训练集中可能某一类别的样本数目过多，从而导致分类器过拟合。本算法是基于 reweighting 的方式对样本权重进行利用，首先对训练集进行 RBWS 重采样，在重采样后更新新训练集的样本权重 $D_t^{'}$ 时，设置新训练集 S_t 中属于原训练集 S 的样本的权重不变，将新样本的权重设为统一值 1/m，m 为初始训练集 S 的样本总数。接下来利用 S_t 和 $D_t^{'}$ 进行基分类器 h_t 的训练，最后计算基分类器 h_t 的伪误差 ε_t，并对权重分布进行更新。经过 T 次迭代训练，得到最终的集成分类器 H。RBWS-ADAM2 算法的伪代码描述如下。

Input:dataset $S=\{(x_1,y_1),?\ ,(x_m,y_m)\}$，$y_i\{1,?\ ,k\}$，$i\{1,?\ ,m\}$ ·

//k 为类别数，m 为样本总数

Output: integrated classifier H.

for each $i\{1,?\ ,m\}$ **do:**

$D_1(i)=1/m$ //初始化样本分布权重

$W_{i,y}^1=D_1(i)/(k-1)$ //初始化权重向量，权重向量 W 的上标 1 代表是第 1 次迭代，i 是样本的序号，y 是样本的类别标签，不包括 y_i，即 $y\neq y_i$

end for

for each $t\{1,?\ ,T\}$ **do:**

for each $i\{1,?\ ,m\}$ **do:**

$W=W+W_i^t$ //求和 $\sum_{i=1}^{m}W_i^t$

end for

for each $i\{1,?\ ,m\}$ **do:**

$q_t(i,y)=W_{i,y}^t/W_i^t$ //计算样本标签权重，其中 $W_i^t=\sum_{y\neq y_i}W_{i,y}^t$，$y\neq y_i$

$D_t(i)=W_i^t/W$ //计算样本分布权重

end for

S_t=RBWS(S) //运用 RBWS 重采样策略对训练集 S 进行预处理得到新训练集 S_t

for each j S_t ? && j∉ S **do:**

$D_t(j)=1/m$ //设置采样过程中产生的新样本权重为 1/m，原有样本的权重保持不变

end for

get $D_t^{'}$ //得到新训练集的样本权重分布 $D_t^{'}$

h_t=Train($D_t^{'}$, S_t) //利用 $D_t^{'}$ 和 S_t 进行基分类器训练，得到基分类器 h_t

for each $i\{1,?\ ,m\}$ **do:**

$\varepsilon_t=\varepsilon_t+D_t(i)(1-h_t(x_i,y_i)+\sum_{y\neq y_i}q_t(i,y)h_t(x_i,y))$ //计算基分类器 h_t 的伪误差 ε_t

end for

$\varepsilon_t=\frac{1}{2}\varepsilon_t$ //得出伪误差 ε_t

$\beta_t=\varepsilon_t/(1-\varepsilon_t)$ //计算 β_t

for each $i\{1,?\ ,m\}$ **do:**

$W_{i,y}^{t+1}=W_{i,y}^t\beta_t^{(\frac{1}{2})(1+h_t(x_i,y_i)-h_t(x_i,y))}$ //更新权重

end for

end for

return $H(x)=\arg\max\sum_{t=1}^{T}\lg\frac{1}{\beta_t}h_t(x,y)$ //T 次循环结束后，输出最终的集成分类器 H，并运用测试样本进行测试，通过投票的方式得到分类结果

2 实验与分析

2.1 实验设置

本文实验采用的两个共享因特网流量数据集分别为 Cambridge1[16]和 Cambridge2[17]。Cambridge1 和 Cambridge2 流量数据集均取自剑桥大学的研究网站，其中 Cambridge1 包括 ENT1，ENT2，…，ENT10 和 ENT12 共 11 个子数据集，Cambridge2 包括 day1，day2，day3 和 siteb 共四个子数据集，本文选取其中 ent1，ent2，day1 和 day2 共四个数据集进行对比实验，数据集 ent1 中各类别样本数目如表 1 所示，可以看出各类别样本数目存在严重的不平衡。

表 1 数据集 ENT1 各类别样本数目

类别	数目	类别	数目
WWW	18211	MAIL	4146
FTP-PASV	43	ATTACK	122
DATABASE	238	FTP-DATA	1319
SERVICES	206	MULTIMEDIA	87
FTP-CONTROL	149	P2P	339

实验中所有参与比较的算法分别为 AdaBoost.M2-reweighting(简称 ADAM2-rewei)，AdaBoost.M2-resampling(简称 ADAM2-resam)，SMOTEBoost(简称 SB-采样比例 100%)，SMOTEBoost(简称 SB-采样比例 200%)，SMOTEBoost(简称 SB-采样比例 500%)，RUSBoost(简称 RB)和本文 RBWS-ADAM2。

实验采用 Python 实现，集成学习算法的迭代次数设置为

100，并选择最大深度为 5 的 CART 决策树作为所有集成学习算法的基分类器。实验结果通过十重交叉验证获得。实验评价分类器性能的指标包括总体分类精度，总体平均 F-measure，总体 G-mean 和单类 F-measure，其中总体平均 F-measure 是所有类别网络流分类 F-measure 的平均值，总体平均 F-measure 和总体 G-mean 都可以衡量分类器对多类不平衡数据集的分类性能。

2.2 实验结果比较与分析

2.2.1 Kappa-error 图

Kappa-error 图是衡量集成分类器多样性的重要表现形式，假设有一个样本数目为 N 的测试集和两个分类器 C1 和 C2，表 2 表示的是测试集中被分类器 C1 和 C2 分类正确或错误的样本数，例如 a 为 C1 和 C2 都分类正确的样本数，b 为 C1 分类正确 C2 分类错误的样本数，abcd 四项的总和为 N。

表 2　C1 和 C2 分类情况示例

	C2 correct	C2 wrong
C1 correct	a	b
C1 wrong	c	d

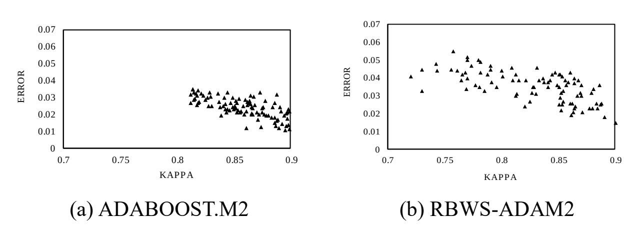
两个分类器之间的差异可以通过 k 值衡量，计算公式如式(1)所示。

$$k=\frac{2(ad-bc)}{(a+b)(b+d)+(a+c)(c+d)} \tag{1}$$

k 值代表的是 Kappa-error 图中 x 轴，值越小代表两个分类器之间差异性越大。此外，可以通过 e 值来衡量两个分类器的平均误差，计算公式如式(2)。平均误差 e 代表的是 Kappa-error 图中 y 轴 error。

$$e=\frac{1}{2}\left(\frac{c+d}{N}+\frac{b+d}{N}\right)=\frac{b+c+2d}{2N} \tag{2}$$

假设一个集成分类器有 L 个子分类器，则这 L 个子分类器就会在 Kappa-error 图中产生 L(L-1)/2 个点，每个点对应一对子分类器。本文计算了 200 个 RBWS-ADAM2 和 200 个 AdaBoost.M2 中子分类器对应的 Kappa-error 点。图 1 中 a 图和 b 图分别是 RBWS-ADAM2 和 AdaBoosT.M2 对应的 Kappa-error 图，每个图中只包含 200 个数据点。



(a) ADABOOST.M2　　(b) RBWS-ADAM2

图 1　Kappa-error 图

如图 1 所示，在 Kappa-error 图 1（b）中，RBWS-ADAM2 的数据点分布相对于图 1（a）的 AdaBoost.M2 在 x 轴和 y 轴方向的跨度都更大，表明 RBWS-ADAM2 的分类器产生了更大的多样性。此外，可以看出图 1（b）中点集分布是倾斜的，在 y 轴跨度相对于图 1（a）更大，表明在增加分类器多样性的同时，分类误差也会略有增大。

2.2.2 实验指标对比分析

表 3~6 是各算法在各数据集上进行对比实验的结果。其中总体分类精度用 ALL-PRE 表示，总体 G-mean 用 ALL-GMN 表示，总体平均 F-measure 用 ALL-AVG-FM 表示，训练时间用 T 表示。

表 3　ent1 各算法对比结果

算法	ALL-PRE	ALL-GMN	ALL-AVG-FM	T(s)
ADAM2-REW	**0.9851**	0.8077	0.8597	146.95
ADAM2-RES	0.9839	0.7651	0.8249	150.06
SB(100)	0.9840	0.7781	0.8444	232.70
SB(200)	0.9744	0.4973	0.7093	213.88
SB(500)	0.9786	0.6672	0.7801	270.97
RB	0.9721	0.8072	0.8267	164.05
本文算法	0.9530	**0.8508**	**0.8997**	608.85

表 4　ent2 各算法对比结果

算法	ALL-PRE	ALL-GMN	ALL-AVG-FM	T(s)
ADAM2-REW	0.9901	0.7952	0.8363	171.17
ADAM2-RES	0.9897	0.8074	0.8543	142.97
SB(100)	0.9889	0.8614	0.9102	203.97
SB(200)	0.9875	0.7785	0.8696	188.23
SB(500)	0.9823	0.7075	0.8372	264.69
RB	0.9889	0.8964	0.9198	170.64
本文算法	**0.9912**	**0.9072**	**0.9369**	590.0

表 5　day1 各算法对比结果

算法	ALL-PRE	ALL-GMN	ALL-AVG-FM	T(s)
ADAM2-REW	**0.9861**	0.8854	0.9001	2850.8
ADAM2-RES	0.9571	0.8989	0.9169	1224.3
SB(100)	0.9546	0.9002	0.8860	5741.2
SB(200)	0.9562	0.9084	0.9083	5023.3
SB(500)	0.9826	0.6667	0.8174	6374.4
RB	0.9809	0.9073	0.8817	2155.2
本文算法	0.9566	**0.9345**	**0.9427**	9798.3

表 6　day2 各算法对比结果

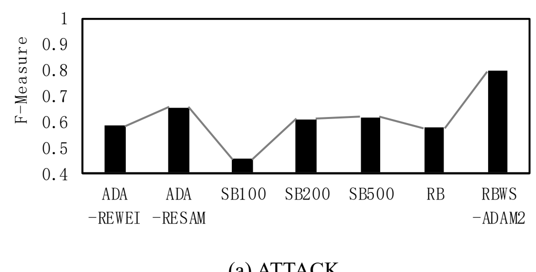
算法	ALL-PRE	ALL-GMN	ALL-AVG-FM	T(s)
ADAM2-REW	0.9333	0.9431	0.9254	3526.3
ADAM2-RES	0.9338	0.9251	0.9114	2686.9
SB(100)	0.9031	0.9384	0.9069	5333.2
SB(200)	0.9201	0.9445	0.9190	6002.2
SB(500)	0.9047	0.9332	0.8564	7956.3
RB	**0.9720**	**0.9588**	**0.9496**	2598.8
本文算法	0.9060	0.9545	0.9316	9853.1

如表 3~6 所示，在数据集 ent1，ent2 和 day1 上，相比其他算法，本文 RBWS-ADAM2 算法在总体 G-mean 和总体平均 F-measure 两个指标上均有较大程度的提升，说明本文算法对于解决集成学习面临的多类不平衡问题，提升 AdaBoost.M2 算法

的整体分类性能起了很好的效果，特别是在数据集 ent2 上，本文算法的总体 G-mean 和总体平均 F-measure 比 AdaBoost.M2-reweighting 提升高达 10%左右。对比总体分类精度，本文 RBWS-ADAM2 算法虽然在数据集 ent2 上最高，但是在其他数据集上均有所下降，与 AdaBoost.M2-reweighting 也基本保持在 0.01~0.04 的差距，这验证了图 1 的分析结论，本文算法对多类不平衡数据分类性能的提升是以损失一定的分类精度为代价的。

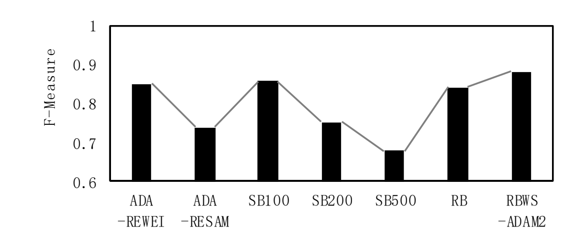
对于 RUSBoost 算法，其在部分数据集上表现很好，但是在其他数据集上出现波动，比如在数据集 DAY2 上该算法的三项指标值都是最高，但是在数据集 ent1 上其总体 G-mean 和总体平均 F-measure 两个指标均与最高值有较大差距。对于 SMOTEBoost 算法，采样比例是影响算法表现的重要因素，本文选取 100%，200%和 500%三个采样阈值进行对比实验的结果表明，不同数据集的最优采样比例可能不相同，例如在数据集 ent2 上，最优的采样比例为 100%，而在数据集 day2 上，最优的采样比例却为 200%。本文 RBWS-ADAM2 算法在所有数据集上的表现都较为稳定，即使在部分数据集中没有取得最好的分类表现，却也保持在较高的水平。

此外，从训练时间的对比可以看出，本文算法的训练时间相比于其他算法较长，出现这种情况是因为本文算法的处理逻辑相比于其他算法更加复杂。为进一步分析本文算法对少数类分类的提升效果，图 2 展示了数据集 ent1 上各算法对各类别的 F-measure 对比结果。

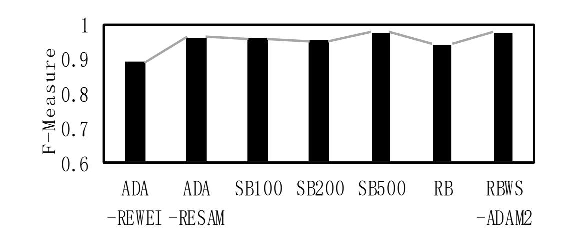


(a) ATTACK

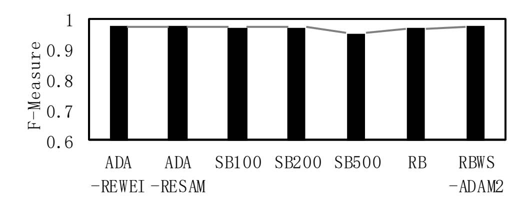
(b) MULTIMEDIA



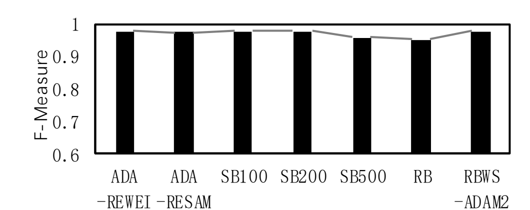
(c) FTP-CONTROL



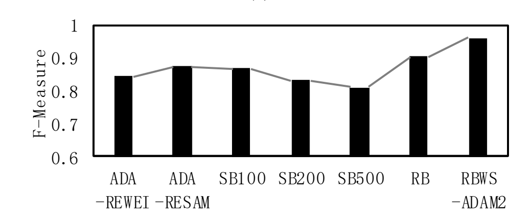
(d) SERVICES



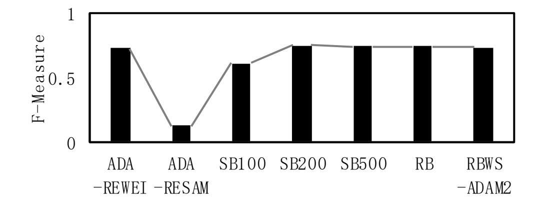
(e) MAIL



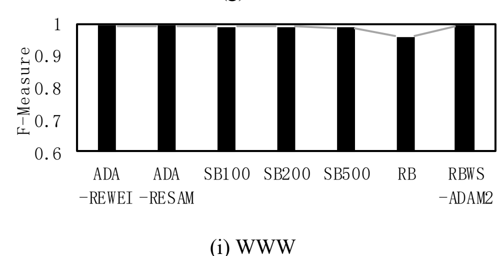
(f) FTP-DATA



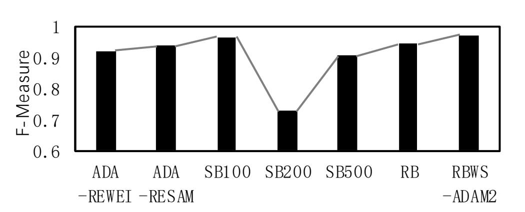
(g) P2P



(h) FTP-PASV



(i) WWW



(j) DB

图 2　ENT1 上单类流 F-Measure 对比

由图 2 可得，本文 RBWS-ADAM2 算法对 Attack，FTP-control，Multimedia 和 P2P 这四类少数类协议的 F-measure 较其他算法均有较为显著的提升，说明本文算法对部分少数类的分类能力相比于其他算法有明显提高，而对其他少数类或多数类，本文算法的 F-measure 也与其他算法中表现最优的基本相近，例如就 DatabasE 而言，SMOTEBoost(200)的 F-measure 最低，而本文 RBWS-ADAM2 算法的 F-measure 与 SMOTEBoost(100)同样处于较高的水平并且差距极小。

就 RUSBoost 而言，对于大部分少数类，其 F-measure 相对于 AdaBoost-reweighting 均有一定程度的提高，而对多数类 WWW，RUSBoost 对应的 F-measure 却略低于其他算法，出现这种情况可能是因为在对多数类随机欠采样时，误删了部分包含有用信息的样本，影响了分类器对多数类的学习，以上分析说明虽然利用随机欠采样来平衡数据集会在一定程度上提升对部分少数类的分类能力，但是很可能会降低对多数类的分类效果。另外从图 2 中 SMOTEBoost 三种不同阈值的实验对比结果可以看出，采样比例对 SMOTEBoost 的算法表现产生了明显的影响。其中对于 FTP-control，Multimedia，Database 和 P2P 四类，F-measure 随着采样阈值的增加不断降低，说明对于这四类，采样阈值不需要超过 100%，对于 FTP-DATA，Mail 和 WWW 三类，三种采样阈值下的算法表现相差不多，而 Attack，FTP-PASV 和 Services 三类的则分别在 500%，200%和 500%取得最高的 F-measure。以上分析表明不仅对不同数据集 SMOTEBoost 算法的最优采样阈值不同，对不同类别的最优采样阈值也可能不同。

此外，从 FTP-PASV 的 F-measure 对比可以发现，AdaBoost-resampling 算法对应的 F-measure 出现了远低于其他算法的情况。通过查看表 1，可以发现在数据集 ent1 中，FTP-PASV 的样本数目仅有 43，远少于其他类别。由此可推断，出现这种情况是因为 AdaBoost-resampling 中采用的有放回随机抽样策略导致数目较少的 FTP-PASV 样本在重采样过程中被忽略，进而造成分类器对该类样本学习不足。

前述实验中，集成学习算法的迭代次数设置为 100，为探究迭代次数对集成分类器分类效果的影响，本文再次设置迭代次数为 0 到 100，利用 AdaBoost.M2 和 RBWS-ADAM2 对数据集 ent1 进行训练与测试，以观察集成分类器的总体平均 F-measure 随迭代次数变化情况，实验结果如图 3 所示。

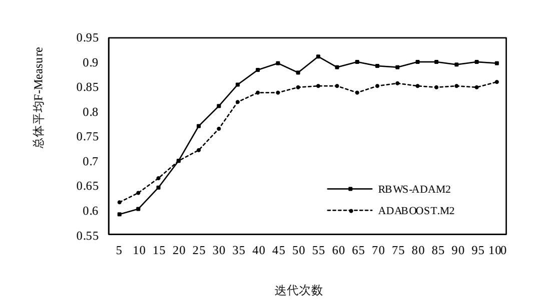


图 3　总体平均 F-measure 随迭代次数变化图

由图 3 可见两种集成分类器的总体平均 F-measure 随着迭代次数增加而不断提升，当迭代次数增加到 40 之后，折线上升的幅度才逐渐变缓，并在数次波动后趋于平缓稳定，最终 AdaBoost.M2 对应的总体平均 F-measure 稳定在 0.85 左右，而本文 RBWS-ADAM2 对应的总体平均 F-measure 稳定 0.90 左右。在折线上升过程中，可以发现当迭代次数约小于 20 时，RBWS-ADAM2 的总体平均 F-Measure 总是低于 AdaBoost.M2，但随着迭代次数增加，RBWS-ADAM2 逐渐高于 AdaBoost.M2 并拉开差距直至最终趋于稳定。可以推断出现这种现象的原因是 RBWS-ADAM2 中构造的训练集具有较大差异性，当迭代次数较低时，其无法代表全部的原始训练集，并且其差异性学习没有发挥优势，还可能因为出现极端差异性而影响分类器的学习效果，导致集成分类器的分类性能甚至比不上 AdaBoost.M2 分类器，然而随着迭代次数增加到一定程度，各训练集在具备差异性的同时，也具备覆盖全部原始训练集的条件，从而保证集成分类器可以充分学习到全部原始训练数据，因此提升了算法的整体性能。

2.3　算法复杂度分析

通过对本文 RBWS 重采样策略的伪代码进行时间复杂度分析，得出第 03 到 07 行的复杂度为 O(k)，第 11 行的复杂度为 $O(kn^2)$，第 12 行到 15 行的复杂度为 $O(kn^2)$，第 18 到 22 行的复杂度为 O(kn)，其他部分复杂度均为常数级。综上，得出 RBWS 重采样策略的复杂度为 $O(n^2)$。

通过对本文 RBWS-ADAM2 算法的伪代码进行时间复杂度分析，得出第 01 到 04 行的复杂度为 O(n)，第 06 到 08 行的复杂度为 O(kn)，第 09 到 12 行复杂度为 O(kn)，第 13 行中 RBWS 重采样过程的复杂度为 $O(kn^2)$，第 14 行到 16 行的复杂度为 O(k)，第 18 行基分类器训练的复杂度为 O(kn)，第 19 到 21 行的复杂度为 O(kn)，第 24 行到 26 行的复杂度为 O(kn)，其他部分复杂度均为常数级。最终得出，本文 RBWS-ADAM2 算法的时间复杂度为 $O(n^2)$。

本文算法和实验中各算法的时间复杂度对比结果如表 7 所示。

表 7　算法时间复杂度对比

算法	复杂度	算法	复杂度
ADABOOST.M2	O(n)	RUSBOOST	O(n)
SMOTEBOOST	O(n)	本文算法	$O(n^2)$

由表 7 可见，本文算法的时间复杂度为 $O(n^2)$，较高于其他算法，这是因为本文算法的处理逻辑相对于对比的算法稍复杂，但从前述实验结果可以看出，本文算法取得了最优的分类效果。

3　结束语

本文主要针对网络流量中的多类不平衡问题，提出一种集成学习算法 RBWS-ADAM2，该算法在 ADABOOST.M2 每次迭

代时，设计了基于样本权重的随机平衡重采样策略来对训练集进行预处理，在解决数据多类不平衡问题的同时，增大训练集之间的差异性以提升集成分类器的泛化性能。在国际公开数据集 Cambridge1 和 Cambridge2 的部分数据集上进行对比实验的结果表明，本文提出的算法不仅可以有效提升集成分类器对于部分少数类的 F-Measure，也有效提升了集成分类器的总体 G-Mean 和总体平均 F-Measure，在缓解数据多类不平衡对少数类分类影响的同时，提高了集成学习算法的整体泛化性能，使得集成分类器在面临多类不平衡网络流量时具备更强的分类能力。后期工作主要是针对本文算法训练时间稍长的不足进行优化。

参考文献：

[1] Khater N, Overill R. Network traffic classification techniques and challenges [C]// Proc of the 10th International Conference on Digital Information Management. 2016: 43-48.

[2] Chawla N, Bowyer K, Hall L, *et al.* Smote: synthetic minority over-sampling technique [J]. Journal of Artificial Intelligence Research, 2002, 16 (1): 321-357.

[3] Yen S J, Lee Y S. Cluster based under-sampling approaches for imbalanced data distributions [J]. Expert Systems with Applications, 2009, 36 (3): 5718-5727.

[4] Agrawal A, Viktor H, Paquet E. SCUT: multi-class imbalanced data classification using smote and cluster-based undersampling [C]// Proc of International Joint Conference on Knowledge Discovery. 2016: 226-234.

[5] Blaszczynski j, Stefanowski j. Neighbourhood sampling in bagging for imbalanced data [J]. Neuro Computing, 2015, 150 (52): 529-542.

[6] Wang S, Yao X. Multi-class imbalance problems: analysis and potential solutions [J]. IEEE Trans on Systems, Man and Cybernetics, Part B, 2012, 42 (4): 1119-1130.

[7] Wang s, Yao x. Diversity analysis on imbalanced data sets by using ensemble models [C]// Proc of IEEE Symposium Series on Computational Intelligence and Data Mining. 2009: 324–331.

[8] Freund Y, Schapire R. Experiments with a new boosting algorithm [C]// Proc of the 13th International Conference on Machine Learning. 1996: 148–156.

[9] Chawla N, Lazarevic A, Hall L. SMOTEBoost: improving prediction of the minority class in boosting [C]// Proc of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases. [S. l.] : Springer, 2003: 107–119.

[10] Barua S, Islam M, Yao X, *et al.* MWMOTE-Majority weighted minority oversampling technique for imbalanced data set learning [J]. IEEE Trans on Knowledge and Data Engineering, 2014, 26 (2): 405-425.

[11] Seiffert C, Khoshgoftaar T, Hulse J V, *et al.* Rusboost: a hybrid approach to alleviating class imbalance [J]. IEEE Trans on Syst, Man Cybern, PartA: Syst. Hum, 2010, 40 (1): 185–197.

[12] Molinara M, Ricamato M, Tortorella F. Facing imbalanced classes through aggregation of classifiers [C]// Proc of the 14th International Conference on Image Analysis and Processing. 2007: 43–48.

[13] Stefanowski J. Overlapping, rare examples and class decomposition in learning classifiers from imbalanced data [M]// Emerging Paradigms in Machine Learning. Springer, 2013: 277-306.

[14] Wang S, Yao X. Relationships between diversity of classification ensembles and single-class performance measures [J]. IEEE Trans on Knowl. Data Eng, 2013, 25 (1): 206-219.

[15] Seiffert C, Khoshgoftaar T, Hulse J V. Resampling or Reweighting: A Comparison of Boosting Implementations [C]// Proc of IEEE International Conference on Tools with Artifical Intelligence. 2008, 1: 445-451.

[16] Moore A M, Zuev D, Crogan M. Discriminators for use in flowbased classification [R]. 2005: 1-16.

[17] Li W, Canini M, Moore A W, *et al.* Efficient application identification and the temporal and spatial stability of classification schema [J]. Computer Networks, 2009, 53 (6): 790-809.